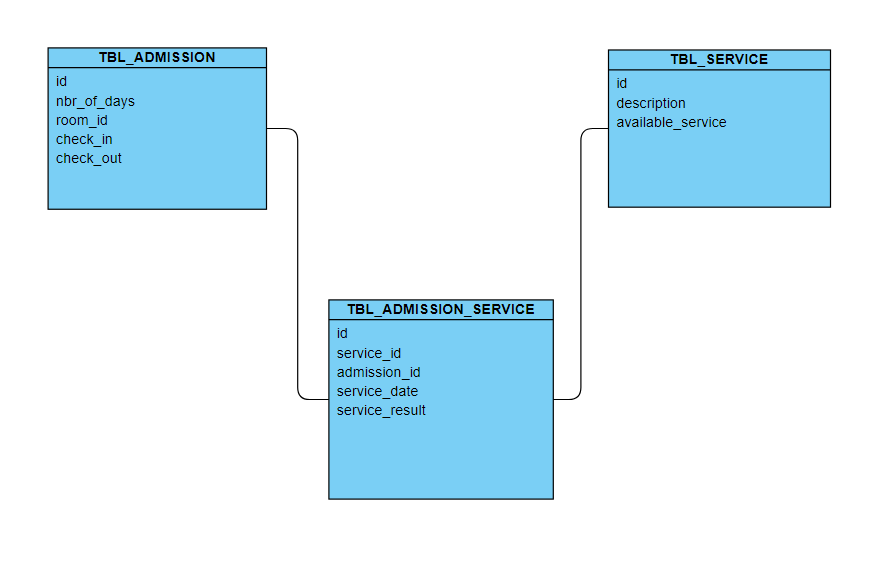
**Micro Service Admission :**

1. **Description Générale :**

Ce micro service a comme but de gérer les admissions de notre hôpital Covid19. Le patient peut s’inscrire pour l’admission s’il veut faire seulement des tests (Test Covid19) ou s’il a besoin de réserver une chambre. En plus, il gère la disponibilité des servies. En faisant une admission, l’utilisateur doit s’inscrire la date d’entrée, la date sortie et les nombres de nuits. D’autre part, pour chaque admission-service, on sauvegarde autre que les infos nécessaires, la date du service et son résultat. Ce micro service fonctionne d’une façon indépendante des autres micro services en communiquant avec sa propre base de donnée.

1. **Diagramme de classe :**



1. **Description des attributs :**

* TBL\_ADMISSION :

Id : Integer : un identifier par incrémentation pour chaque admission.

Nbr\_of\_days : Integer : le nombre de jours d’admission.

Room\_id : Integer : un identifier indiquant la chambre réservée.

Check\_in : Date : la date entrée.

Check\_out : Date : la date de sortie.

* TBL\_SERVICE :

Id : Integer : un identifier par incrémentation pour chaque service.

Description : String : la description du service.

Available\_service : Boolean : si le service est disponible à l’hôpital ou non.

* TBL\_ADMISSION\_SERVICE :

Id : Integer : un identifier par incrémentation pour chaque admission-service.

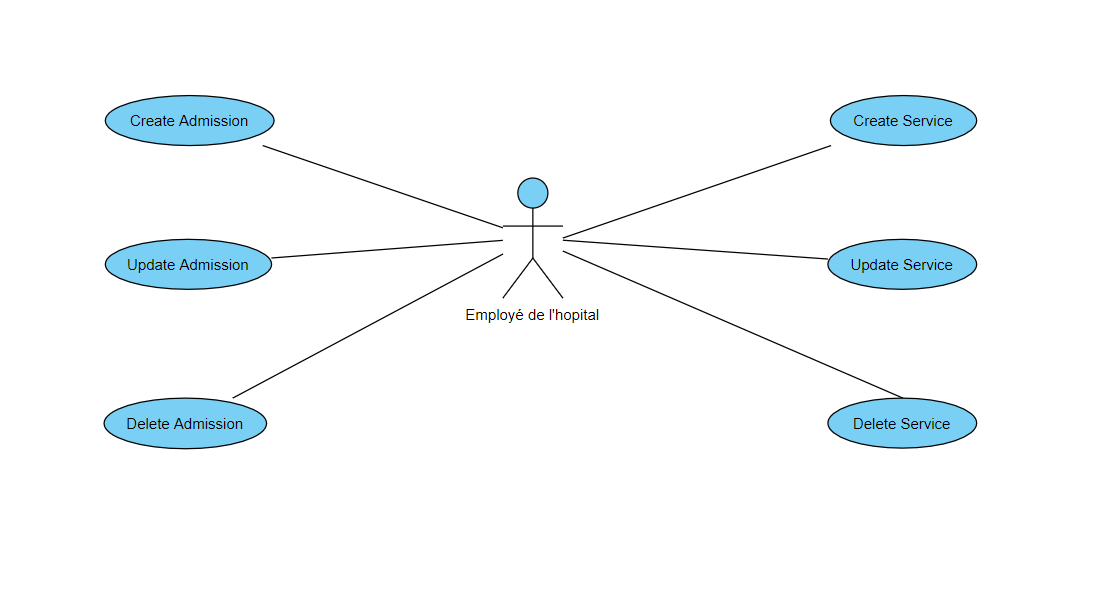
Service\_id : Integer : un identifier indiquant le service.

Admission\_id : Integer : un identifier indiquant l’admission.

Service\_date : Date : la date de la réalisation du service.

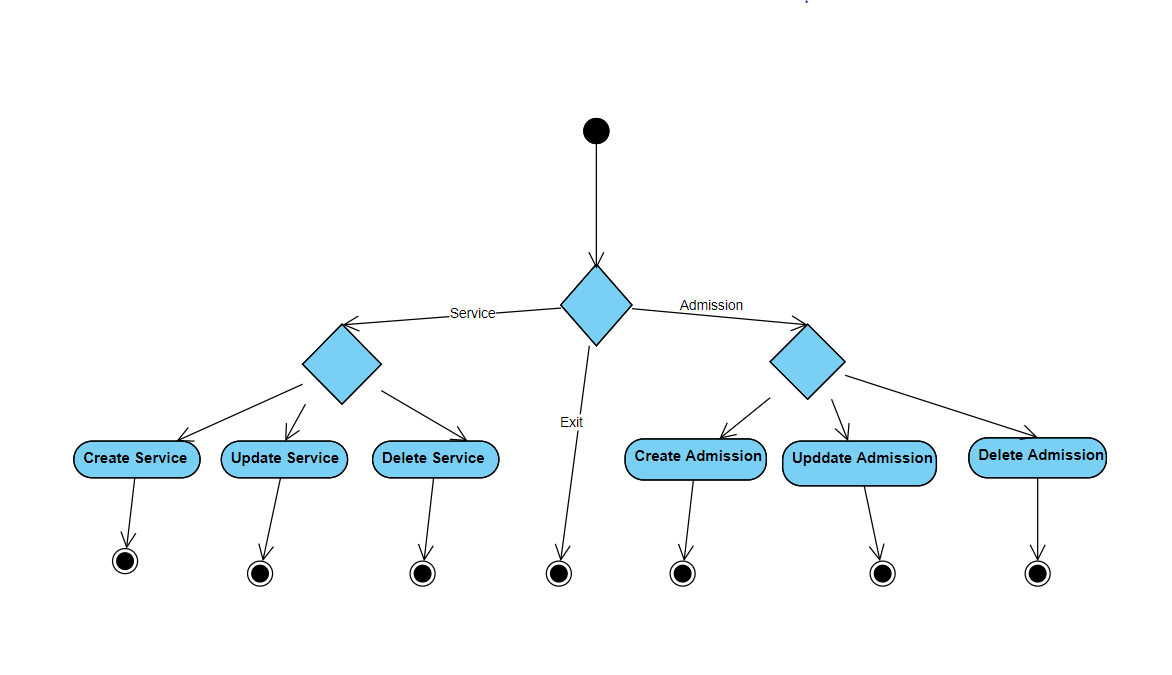
Service\_result : String : la description du résultat du service.

1. **Cas d’utilisation :**



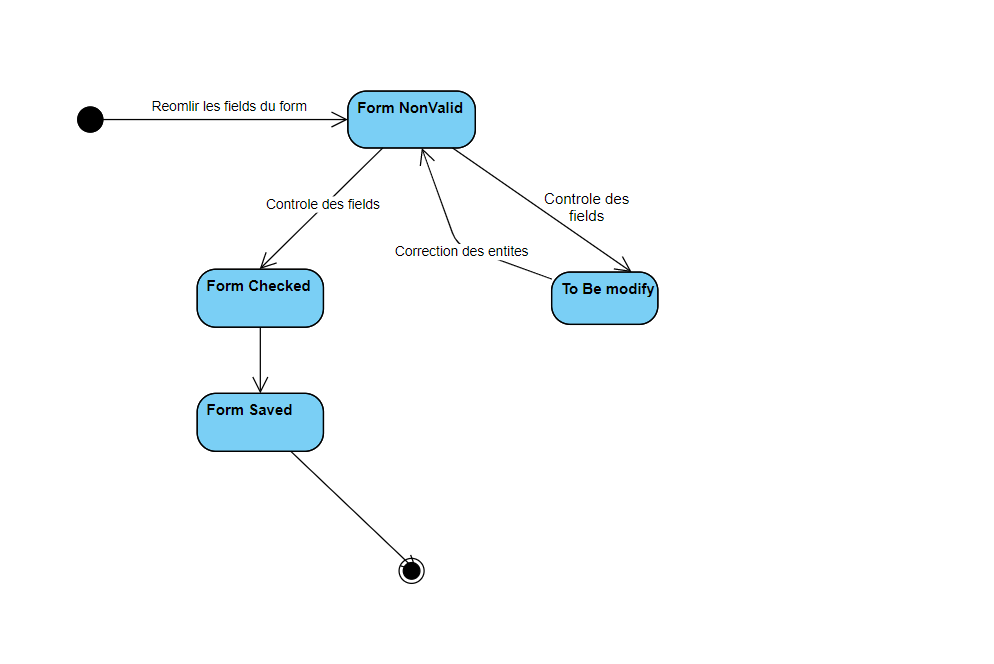
Puisque c’est un micro service donc l’employée tout seul peux subir plusieurs actions d’une façon indépendante des autres micro services. Les actions sont : Create/Update/Delete Service et Create/Update/Delete Admission.

1. **Diagramme d’activité :**



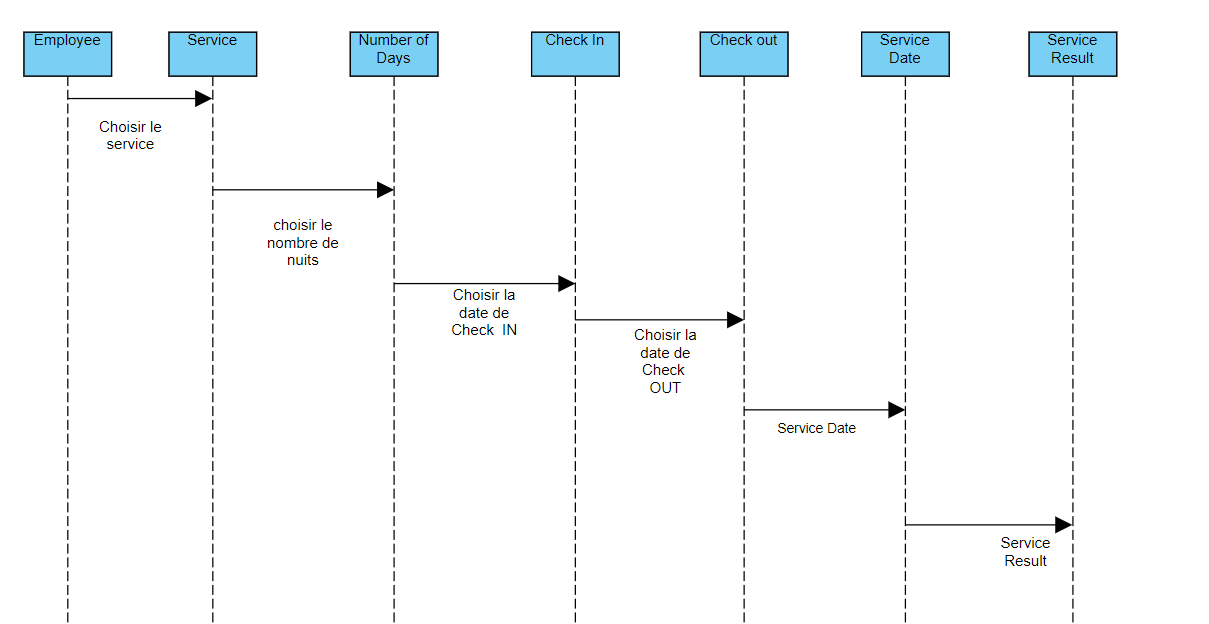
Après choisir ou il faut faire le travail (service ou application), l’employé peux subir les Crud Operations (Create ou Update ou Delete) pour un service ou pour une admission et il quitte.

1. **État-transition :**



La forme de l’admission comme la forme du service contient des contrôles et des règles suivant les besoins. Donc, lorsque les Fields sont remplis la forme devient NonValid, puis si les Fields respectent les règles déjà mis la forme devient forme Checked sinon elle devient To Be Modify et elle retour à la forme non valide pour corriger les erreurs. Après que la Forme est checked, la forme est enregistrée sous forme Saved et il quitte.

1. **Diagramme de séquence :**



Pour faire une admission, il faut choisir le service (Si le service est indisponible il sera invisible dans la liste des services), le nombre de nuits, la date de Check In, la date Check Out (ce Field peut être remplie dans l’update admission), la date de service, le résultat de service (ce Field peut être remplie dans l’update admission).

1. **Micro Services Fonctions :**

* **Admission**

GetAdmissionEntityById : Récupérer les informations d’une admission suivant un ID.

CreateAdmissionEntity : Créer une admission en remplissant tous les informations nécessaires.

UpdateAdmissionEntity : Update les informations d’une admission, par exemple insérer la date de sortie (Date Check Out).

DeleteAdmissionEntity : Supprimer une admission.

* **Admission-Service**

GetAdmissionServiceEntityById : Récupérer les informations d’une admission-service suivant un ID.

CreateAdmissionServiceEntity : Créer une admission-service en remplissant tous les informations nécessaires.

UpdateAdmissionServiceEntity : Update les informations d’une admission-service, par exemple insérer le résultat d’un service.

DeleteAdmissionServiceEntity : Supprimer une admission-service.

* **Service**

GetServiceEntityById : Récupérer les informations d’un service suivant un ID.

CreateServiceEntity : Créer un service en remplissant tous les informations nécessaires.

UpdateServiceEntity : Update les informations d’un service, par exemple modifier la disponibilité d’un service.

DeleteServiceEntity : Supprimer un service.